실내 주행 이동 로봇을 위한 컬러 이미지를 이용한 표식점 위치 측정 방법

論 文
45~2~19

A Landmark Position Estimation Method Using a Color Image for an Indoor Mobile Robot

俞 元 弼* · 鄭 明 振**
(Won-Pil Yu · Myung-Jin Chung)

Abstract - It is very important for a mobile robot to estimate its current position. With precise information about the current position, the mobile robot can do path-planning or environmental map building successfully. In this paper, a position estimation method using one color image is presented. The mobile robot(K2A) takes an image of a corridor and searches for the door and pillar, which are the given landmarks. The color information is used to distinguish the landmarks. In order to represent the presence of the landmarks, Image Mode is defined. This method adopts Kullback information distance. If a landmark is detected, with the color information, the mobile robot identifies the vertical line of the landmark and its crossing point with the floor. After finding the crossing point, the mobile robot calculates the distance from its current position to the crossing point and an experimental navigation is performed.

Key Words : Mobile Robot, Color Information, Landmark, Kullback Information Distance, Image Mode

1. 서 론

이동 로봇의 위치 추정 문제는 외부 환경을 인식하고 이로부터 자신의 위치를 파악하는 것으로 여러 가지 다양한 센서들을 통하여 해결할 수 있는데 대표적인 것으로 시각 센서를 들 수 있다. 시각 센서의 경우 공간적인 형태에 대한 정보를 에지(edge) 처리를 이용하여 쉽게 얻을 수 있고 근거리에 대한 위치 정보를 얻기가 쉽기 때문에 실내에서 표식점(landmark)을 이용하여 로봇의 위치를 추정하는 문제 해결에 적합하다. 하지만, 시각 센서의 경우 처리해야 할 데이터량이 많기 때문에 실시간 처리를 하기 위해서는 고속의 프로세서가 필요하게 된다. 또한, 이동 로봇 주행의 관점에서 볼 때 표식점이 주어져 있다고 하더라도 이미지 상의 표식점의 위치를 파악하거나 표식점의 존재 유무를 판단해야 하는 문제가 생기며 기존의 흑백 이미지를 이용하여 표식점을 추출하는 경우에 주위와 대비되는 명암과 모양을 가지지 않으면 감지하기 어려운 단점이 있다. 또한 시각센서의 경우 3차원 물체를 2차원 평면으로 투사시키는 것이기 때문에 3차원상의 좌표값을 한 장의 이미지로는 알 수가 없게 된다.

두 대의 카메라를 이용하는 기존의 stereo vision 방법은 두 카메라의 떨어진 거리를 알고 있고 목적 물체의 이미지상의 위치를 알고 있다는 가정아래 3차원상의 위치 정보를 알 수 있다[1]. 따라서 stereo vision의 경우 두 개의 이미지상에서의 동일 물체를 찾아내는 correspondence 문제를 해결해야 한다[2]. 한 대의 카메라로부터 얻어진 영상만을 가지고 위치 정보를 얻고자 하는 경우 stereo vision의 경우와 같은 correspondence 문제는 없어지지만 3차원 정보를 2차원 정보로부터 추정해내는 것은 불가능하다. 하지만 특정 물체의 3차원 정보 중 어느 하나만 알고 있다면 이것으로부터 나머지 2개의 알려져 있지 않은 정보를 알 수 있다[3].

한 장의 이미지로부터 이동 로봇의 위치를 추정하는 방법은 결국 위의 이유때문에 아주 제한적일 수 밖에 없게 된다. 하지만 실내의 경우 바닥면이 비교적 평탄하다고 할 때 바닥면 위에 놓여져 있는 물체의 위치는 그 물체의 2차원상의 좌표로부터 구할 수가 있다. 예를 들면, 한 대의 카메라로부터 이동로봇의 3자유도 정보를 얻기위한 방법으로 영상 이미지에 맺힌 여러 개의 표식점들의 위치로부터 로봇의 현재 위치를 추정하는 방법[4], 실내 복도의 경우 복도의 수평선과 수직선으로부터 소실점(vanishing point)을 구하고 이로부터 문이나 벽 등에 설치된 물체의 위치를 구하는 방법을 제시한 경우가 있다[5]. 여러 개의 표식점들을 감지하고 이로부터 로봇의 위치를 추정하는 방법은 이미 알려진 특정 물체의 위치와 형태를 로봇이 알고 있어야 하며 Sugihara[4]의 경우도 표식점을 2차원 공간상의 점으로 취급하여 표식점의 공간적인 배치로부터 이동 로봇이 위치할 수 있는 지점의 가능성 정도를 조사하는 방법을 사용하고 있어 여러 개의 표식점들에 대한 correspondence 문제를 해결해야 하며 실제로 3차원상에서 발견되는 일상적인 물체를 표식점으로 이용하는 경우에 대하여 표식점을 추출하는 방법에 대한 제안을 하지 않고 있다. 실내 특히, 복도 주행 문제에 있어서 흔히 사용되는 소실점을 이용한 방법은 실내에서 발견되는 수직선과 평행선을 찾는 방법이 상당히 시간이 걸리고 카메라의 캘리브레이션이 상당히 잘 되어있어야하므로 이동 로봇처럼 주행하면서 카메라의 회전이나 로봇의 전진, 회전 동작에 의한 조정 정도의 변화가 생기는 경우 정확한 위치 정보를 얻

* 正 會 員 : KAIST 電氣 및 電子工學科 博士課程
** 正 會 員 : KAIST 電氣 및 電子工學科 敎授 · 工博
接受日字 : 1995年 4月 26日
1次 修正 : 1995年 8月 21日
2次 修正 : 1996年 1月 8日

기가 어렵게 된다.

컬러 이미지 처리에 대한 연구는 인간의 컬러 정보 처리 과정에 착안하여 이를 기존의 이미지 처리 기법에 이용하는 방법이 있다[6,7]. 실제로 인간의 컬러 정보 처리 과정은 아직 완전하게 알려져 있지 않기 때문에 실제로 사용하는 경우에는 이를 가능한 한 비슷한 형태로 모방하여 적용하는 경우가 많다. 표식점을 감지하기 위한 방법으로 컬러를 이용한 경우는 아직 그렇게 많지 않으나 특정 물체를 표현하는 색상을 몇개의 기본 함수의 합으로 표현하여 비교하는 방법 등이 나와 있고 여러가지 다른 색상 모델에 대하여 어떤 모델이 기존의 이미지 처리 방법에 가장 유용한지를 비교한 경우가 있다[6,7]. 실제의 3차원 물체는 여러가지 다양한 색상 정보를 가지고 있기 때문에 이를 이용하면 물체 인식에 매우 유용하다. 인간의 색상에 대한 시각 처리 기작에 관한 연구는 많이 있었지만 아직 이동 로봇의 주행 문제에 적용한 경우는 적기 때문에 이 분야에 대한 연구가 필요하다고 생각된다.

본 연구에서는 한 대의 컬러 CCD 카메라를 이용하여 표식점으로 주어진 물체의 색상 정보를 영상에서 얻어진 정보와 비교하여 표식점을 감지하는 방법을 제시하고자 한다. 이 때의 표식점은 임의적인 물체를 추가하지 않고 복도에서 흔히 볼 수 있는 문이나 문틀 등을 이용하며 이러한 표식점들을 감지함으로써 하나의 이미지로부터 표식점의 위치 정보를 얻을 수 있음을 보인다. 또한, 위치 정보를 얻기까지의 과정을 색상 정보를 이용하여 구현할 수 있음을 보이며 여러 개의 표식점들을 색상 정보를 이용하여 쉽게 구분할 수 있음을 보이고자 한다.

2. 시스템 구성

그림 1의 실험에 사용된 로봇의 모습을 나타내었다. 이동 로봇은 synchro-drive 방식을 취하고 있어 제자리에서 회전이 가능하며 호스트인 386PC와는 직렬 포트를 이용하여 명령을 받게 된다. 이미지 그래버는 내부에 세 개의 512×512×8 bit 크기의 버퍼를 가지고 있으며 각각 R(ed),G(green),B(lue) 혹은 H(ue),S(aturation),I(ntensity)의 형태로 컬러 이미지를 저장할 수 있다. 카메라는 pan, tilt의 움직임이 가능하며 각각 360°, 60° 크기의 동작을 취할 수 있으며 0.9° 의 분해능을 가진다.

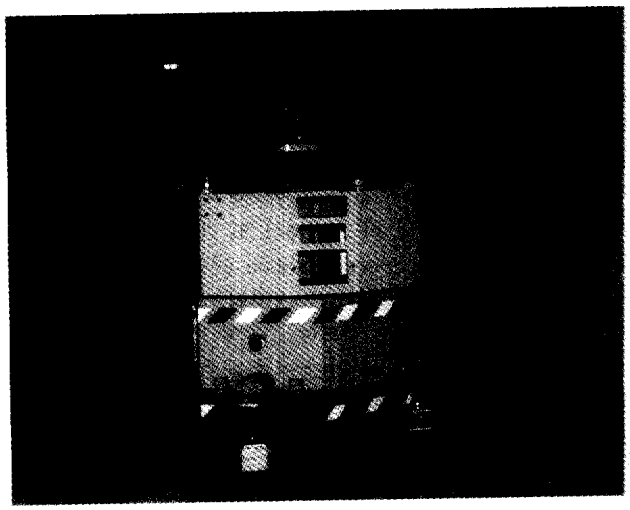

그림 1 이동 로봇의 모습
Fig. 1 The picture of the mobile robot

3. 컬러 비전 모델

임의의 컬러는 3가지의 기본 요소(primary color)의 합으로 표현할 수 있으며 이 때의 기본 요소가 무엇인가에 따라 여러가지 다른 형태의 컬러 모델이 존재하게 된다[8]. 본 절에서는 RGB 컬러 모델과 실험에 도입된 HSI와의 비교를 통하여 HSI 컬러 모델의 장점을 소개하고자 하며 이것은 다른 컬러 모델과의 비교에서도 적용될 수 있는 HSI 컬러 모델의 특성이기도 하다. 주어진 임의의 자연색을 표현함에 있어 특정 컬러 모델에 따라 표현하지 못하는 컬러가 있을 수 있지만 RGB 컬러 모델의 경우, 임의의 색상을 R,G,B 세 가지 성분의 합으로 표현 가능하다는 장점을 가진다. 하지만, RGB 컬러 모델은 각 성분간의 correlation이 매우 크기 때문에 기존의 이미지 처리에 적합하지 못하다. 예를 들어 intensity 성분에 대하여 에지 처리를 하고자 하는 경우, intensity 성분은 RGB 컬러 모델의 경우 식(1)과 같이 표현되므로 각각의 성분에 대하여 에지처리를 해야 하는 단점을 가지며 R,G,B 중 어느 하나의 성분값이 바뀌는 경우 결과를 예상하기가 어렵다. 이에 비해 HSI 컬러 모델은 3가지 단위 중 하나가 이미 intensity 성분이기 때문에 기존의 흑백 이미지에서의 에지처리를 나머지 두 성분과 무관하게 취할 수 있으며 각각의 성분의 변화에 의한 결과를 충분히 예상할 수 있기 때문에 이미지 처리의 관점에서 매우 편리하다고 말할 수 있다. 한편, 컬러 모델을 도입한 자체의 이유는 이미 주어진 여러 개의 표식점들의 존재 유무와 분류라는 문제를 해결함에 있어 기존의 흑백 이미지만을 가지고 작업을 수행할 경우 표식점 자체가 구별하기 쉬운 형태를 가지지 않는한 이미지 해석상의 관점에서 볼 때 각각의 영상 이미지에 대하여 비교적 일반적인 알고리듬을 구현한다는 것은 매우 어렵기 때문이다. 예를 들어, 실내의 경우 비교적 찾기 쉬운 수직선 성분만을 고려한다고 할 때도 각각의 수직선이 실제 공간상의 어떤 물체에 해당하는 지를 판단하는 경우 주변의 에지 성분들을 기초로 상당한 해석을 해야 하는 어려움에 빠지게 된다. 이에 비해 물체 고유의 색 정보를 이용함으로써 알고리듬의 간편화를 꾀할 수 있으며 이미지 해석에서의 자유도를 늘일 수가 있다. 식(1)은 실험에 사용된 HSI 컬러 모델의 기본 요소들을 표현하는 식이다.

$$I=\frac{R+G+B}{3},\ S=1-\frac{\min(R,G,B)}{I},$$
$$H=\frac{1}{360}\left[90-\tan^{-1}\left(\frac{2R-G-B}{\sqrt{(G-B)}}\right)+[0,G>B;180,G<B]\right] \quad (1)$$

4. 컬러 이미지 해석

이동 로봇이 주행하는 경우 가장 문제가 되는 로봇의 현재 위치와 주변 환경과의 공간적인 관계를 알아 내는 문제를 미리 표식점에 대한 정보가 주어졌다고 가정했을 때, 표식점을 발견해내고, 로봇으로부터의 위치를 계산하는 것이 본 실험의 주 내용이다. 본 절에서는 실제 복도상의 한 지점을 감지하기 위

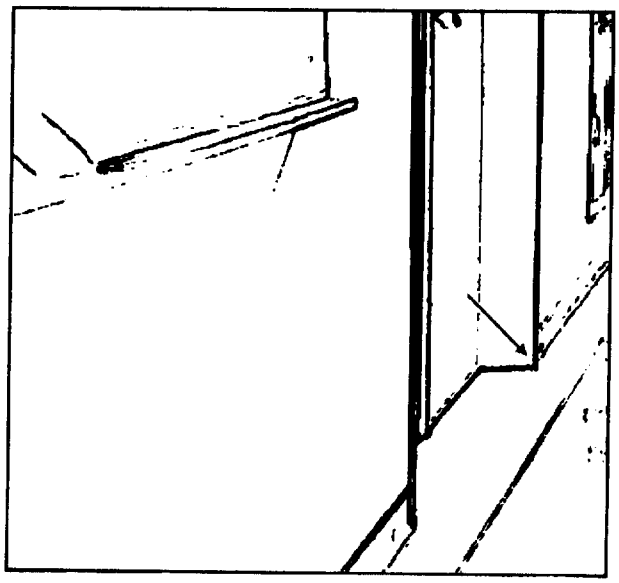

그림 2 복도의 이미지
Fig. 2 The camera image of the hallway

한 알고리즘을 소개하고자 한다. 그림 2는 표식점으로 주어진 문과 문틀의 모습을 intensity 이미지를 이용하여 나타낸 것이다. 본 실험에서 로봇의 관심이 되는 지점은 그림 2의 경우에 화살표로 표시된 지점이며 로봇은 현 위치로부터 해당 지점의 위치를 한 장의 컬러 이미지를 이용하여 계산하고 복도 상에서 벽으로부터의 위치와 표식점까지의 거리를 파악, 주행에 필요한 로봇 명령을 생성하게 된다. 이것을 해결하기 위해서 선행되어야 할 것은 크게 다음의 두 가지를 들 수 있다.

(1) 표식점(문과 문틀)이 발견되었는지의 판단문제
(2) 표식점의 이미지 상에서의 좌표값을 알아내는 문제

4.1 Image Acquisition

한 대의 RGB 카메라로부터 얻어진 영상은 컬러 이미지 그래버(DT-2871)로 보내져 A/D 변환되고 이것은 다시 RGB/HSI 변환을 거치게 된다. 이 과정은 이미지 그래버 내부의 두 개의 CMOS ASIC에 의해 실시간(conversion rate:10MHz)으로 이루어지며 내부의 3개의 버퍼에 H, S, I의 형태로 저장되게 된다.

4.2 전처리 과정

원래의 영상 이미지는 주위의 노이즈나 영상 이미지에 대한 샘플링, 양자화등의 결과로써 생기는 여러 가지 형태의 잡음원을 포함하고 있다. 이것은 개별적인 픽셀(pixel)의 양자화 값에 영향을 미치게 되나 공간적으로는 uncorrelated 되어 있으므로 low pass filtering으로 영향을 줄일 수 있다. 본 실험에서는 원래의 영상 이미지에 3×3 크기의 low pass 마스크를 취하여 이러한 잡음을 줄였고 Sobel 마스크를 이용하여 이후의 실험에 사용될 에지 성분을 추출하였다. 실험에서 위치를 측정하기 위한 표식점은 문틀의 외곽선과 복도 바닥면이 만나는 지점이 된다. 이미지가 문과 문틀을 포함하고 있는지를 구분하기 위해서는 전체 이미지를 수직선으로 몇 개의 영역으로 구분하여 각 영역에 대하여 색 분포를 조사하는 것이 간단하다. 여기에 쓰이는 수직선은 에지 처리 후의 외곽선을 이용하게 된다. 한편, 에지 처리 후의 각 픽셀의 값은 0 ~ 255사이에 분포되어 있기 때문에 실제로 어느 값 이상을 에지라고 판단할 것인가의 문제가 생긴다. 본 실험에서는 이 문제를 auto-thresholding 기법[9]을 이용하여 해결하였으며 thresholding 이후에 얻어지는 이진 영상에서 수직선을 찾게 된다. 한편, 바닥면의 굴곡에 의한 임펄스 잡음과 벽면의 에지 근처의 잡음이 아직 남아 있기 때문에 이것을 최소화하기 위하여 median filter를 취하였다. Median filter는 그 특성상 주변의 값(3×3)에서 중앙값을 선택하기 때문에 특성이 다른 데이타를 제거해 내는 효과가 뛰어나다. 또한, 앞단의 thresholding 처리에 의해 이진 영상으로 바뀌어져 있는 상태이기 때문에 median filter를 구현하기 쉽고 처리 속도도 빨라지게 된다.

4.3 이미지 해석

Segmentation을 이미지 상에서 서로 비슷한 성질을 가지는 부분들끼리 grouping하는 것이라 할 때, segmentation은 본 실험의 경우에 표식점으로 주어진 문과 문틀을 각각 하나의 그룹으로 구분짓고자 하는 것이 목적이 되며, 이것은 표식점인 문과 문틀은 이미지 상에서 한 부분을 차지하고 있기 때문에 색상 정보를 이용하여 표식점이 감지되었는지를 알아내는데 segment 별로 처리하는 것이 편하기 때문이다. 이것은 전처리 과정에서 얻어진 에지 성분으로부터 수직선을 찾고 이로부터 이미지를 몇 개의 구획으로 나눔으로써 가능하다. 각각의 구획 내에 표식점이 존재하는지 아닌지는 컬러 정보를 이용하여 판단하게 된다. 수직선을 찾는 알고리즘은 매우 간단하다. 즉, 종방향으로 픽셀 수가 미리 정해진 값 이상인 경우에 이 방향으로 수직선이 있다고 판단하며 이때의 횡방향의 픽셀 좌표값을 저장한다. 본 실험의 경우 종방향의 픽셀수가 200이상일 경우 수직선으로 인식하였다. Segment 별로 나누는 경우에 저장된 수직선의 횡방향 픽셀 좌표간의 간격이 40이상이 되면, 하나의 segment로 인식하도록 하였다. 따라서, 문틀과 같이 한 표식점 내에 여러 개의 수직선이 존재하는 경우에 이것을 하나의 segment로 구분할 수 있다.

4.3.1 표식점의 컬러 정보

그림 3에 표식점으로 쓰인 문과 문틀, 그리고 복도와 벽의 hue, saturation 값의 분포를 나타내었다. Hue 값의 분포를 보고 알 수 있듯이 실험 장소인 복도는 각 영역의 hue 값이 매우 유사함을 알 수 있다. 따라서, 문제는 비슷한 hue 분포를 가지는 영역을 saturation 값을 이용하여 구분해 내는 것이 된다. 각 영역의 구분을 잘 하기 위해서 또 다른 thresholding 기법을 적용하였다. 이것은 segmentation 작업에서 구분된 영역이 실제로 벽과 문이 한 영역으로 구별될 수도 있기 때문에 서로 다른 분포를 가지는 두 정보를 추출해 내는 문제로 귀착된다. 그림 3의 saturation 정보의 분포를 보면 관심 대상인 문과 문틀의 saturation 값이 200 이상의 분포가 많음을 볼 수 있다. 따라서, 벽과 문 혹은 문틀과 벽, 벽, 문, 문틀이 함께 혼재하게 되더라도 200이상의 saturation 값만 추출할 수 있으면 어떤 이미지가 보여지고 있는지 판단이 가능하게 된다.

4.3.2 Kullback Information Distance

앞 절에서 언급한대로 이미지 상에 혼재하고 있는 표식점의

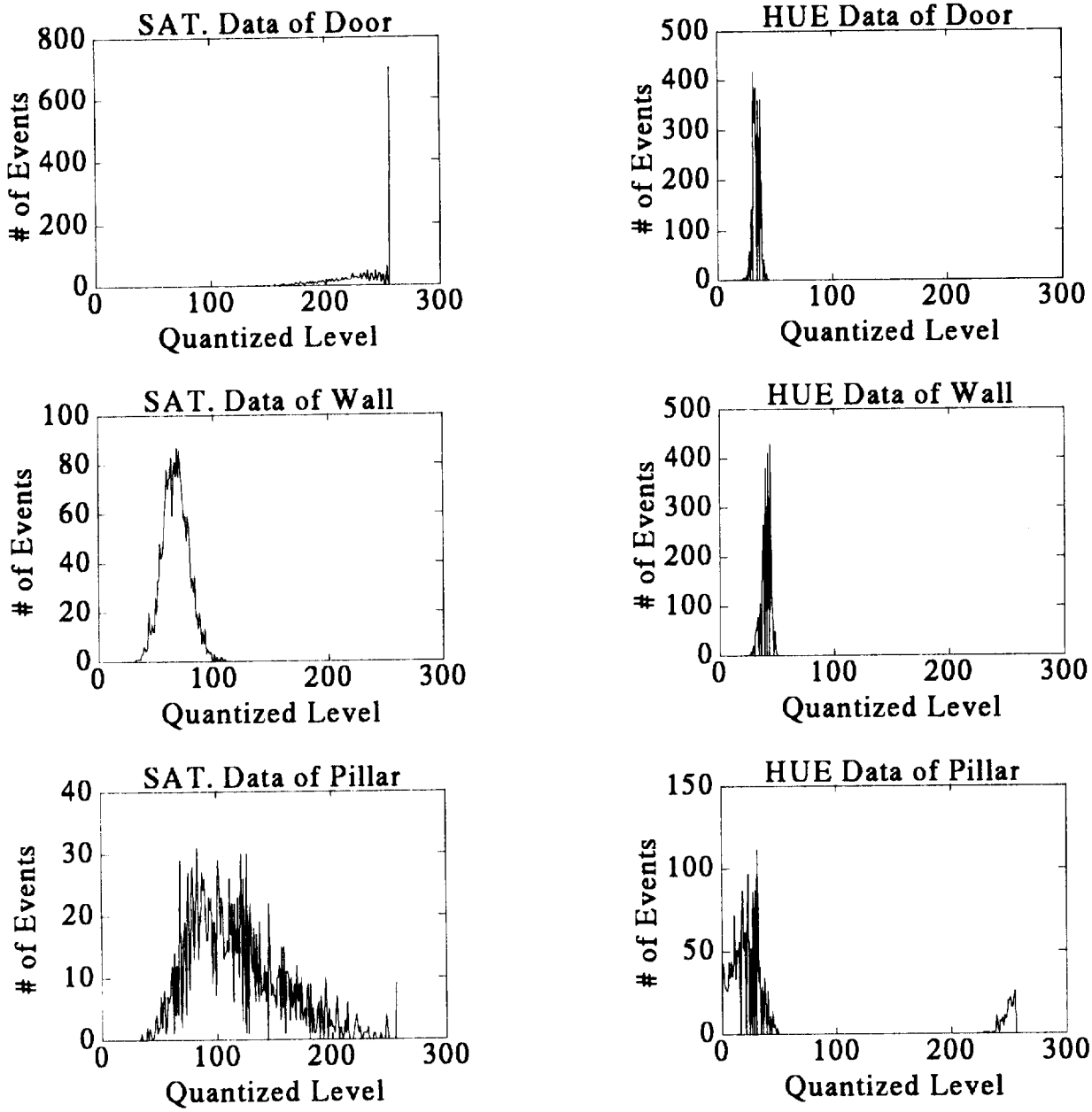

그림 3 표식점의 hue, saturation 값의 히스토그램

Fig. 3 The Histograms for hue and saturation data of the landmarks

위치를 알아내기 위해서 Kullback information distance를 도입하였다[10]. 이것은 일종의 thresholding 기법으로 다음의 가정으로부터 출발한다.

(1) 관찰되는 데이타의 분포는 (μ_1, σ_1^2), (μ_2, σ_2^2)의 분포를 가지는 두 개의 gaussian 함수로 구성되어 있다고 가정한다.

(2) 각각의 구성 비율은 q_1, q_2이다.

(3) Thresholding 값은 아래 식의 Kullback divergence J를 최소화시키는 것 으로 결정된다.

$$J=\sum_{i=0}^{I} P(i)\log\left[\frac{P(i)}{f(i)}\right]=\sum_{i=0}^{I} P(i)\log P(i)-\sum_{i=0}^{I} P(i)\log f(i) \quad (1)$$

이 때 $P(i)$는 히스토그램의 결과로써 얻어진 값이며 알려지지 않은 분포를 가지고 있는 $f(i)$는 분포 h_1의 구성 비율이 q_1, 분포 h_2의 구성 비율이 q_2라고 했을 때, 식(2)와 같이 나타낼 수 있다.

$$f(i)=q_1h_1(i)+q_2h_2(i) \quad (2)$$

한편, 식(1)의 첫 번째 오른쪽 항의 log 값은 물리적으로 다음과 같이 해석할 수 있다. 즉, 어떤 사건 i가 발생하였을 경우, 알려지지 않은 분포 $f(i)$에 대한 정보가 많이 알려져 있을수록 정보량은 작아지므로 $\log[P(i)/f(i)]$는 역으로 $P(i)$가 실제로 알려지지 않은 분포 $f(i)$를 얼마나 잘 묘사할 수 있는가를 나타낸다고 할 수 있다. 따라서, 식 (1)을 최소화시키기

위해서는 식 (1)의 오른쪽 항 $\sum_{i=0}^{I} P(i)\log P(i)$은 상수이므로 두번째 항을 최소화시키는 i값을 구하여야 한다. 이것을 H로 나타내면,

$$H = -\sum_{i=0}^{I} P(i)\log f(i) \tag{3}$$

두 개의 분포가 thresholding 값에 의해 잘 구분되었을 경우, H는 가정 1로부터 아래와 같이 나타낼 수 있다.

$$\begin{aligned} H(t) = 1 &\sum_{i=0}^{t} P(i)\log \frac{q_1(i)}{\sqrt{2\pi\sigma_1(i)}}\, q_1(i) \\ &\exp\left[1\frac{1}{2}\left(\frac{i-\mu_1(i)}{\sigma(i)}\right)^2\right] \\ &-\sum_{i=t+1}^{I} P(i)\log -\frac{q_2 Ii)}{\sqrt{2\pi\sigma_2(i)}}\, q_2(i) \\ &\exp\left[-\frac{1}{2}\left(\frac{i-\mu_2(i)}{\sigma_2(i)}\right)^2\right] \end{aligned} \tag{4}$$

위 식을 간략화하면 식 (5)의 결과식을 얻게 된다.

$$\begin{aligned} H(t) = &\frac{1+\log 2\pi}{2} - q_1(t)\log q_1(i)\log q_1(t) \\ &-q_2(t)\log q_2(t) + \frac{1}{2}(q_1(t)\log \sigma_1^2(t) \\ &+q_2(t)\log \sigma_2^2(t)) \end{aligned} \tag{5}$$

이제 남은 일은 H를 최소화시키는 t값을 찾는 것이 되며, 이 때의 t값이 구하고자 하는 thresholding 값이 된다. 본 실험에서는 식 (5)를 이용하여 이미지 내에서의 표식점 유무를 결정하였다. 이 때, $q_1(t)$, $q_2(t)$, $\mu_1(t)$, $\mu_2(t)$, $\sigma_1(t)$, $\sigma_2(t)$는 각각 다음의 식으로부터 구해진다.

$$q_1(t) = \sum_{i=0}^{t} P(i),\ q_2(t) = \sum_{i=t+1}^{I} P(i) \tag{6}$$

$$\mu_1(t) = \sum_{i=0}^{t} iP(i)/q_1(t),\ \mu_2(t) = \sum_{i=t+1}^{I} iP(i)/q_2(t) \tag{7}$$

$$\begin{aligned} \sigma_1^2(t) &= \sum_{i=0}^{t}[i-\mu_1(t)]^2 P(i)/q_1(t), \\ \sigma_2^2(t) &= \sum_{i=t+1}^{I}[i-\mu_2(t)]^2 P(i)/q_2(t) \end{aligned} \tag{8}$$

이 때 I는 버퍼의 데이타 크기를 나타내며 255의 값을 가지게 된다. H를 최소화시키는 thresholding 값을 구하고 이것으로부터 표식점의 존재 유무를 판단하게 된다.

5. 실험 결과 및 분석

5.1 이미지 처리 결과

본 절에서는 앞서 설명한 전 과정을 실제의 복도 이미지에 적용시켜 이미지에서 표식점이 감지되었는지를 판단하는 것을 보이고자한다. 이를 위해 Image Mode라는 용어를 고안하였다.

• Image Mode

한 장의 이미지를 대표하는 숫자 값으로서 정량적인 크기를 이용하여 표식점의 유무를 판단하게 된다. Image Mode 값을 구하는 과정은 다음과 같다.

(1) 각각의 segment에 대하여 식 (5)를 이용하여 Kullback information distance를 최소화하는 값을 구한다. 이 때 사용되는 saturation 데이타는 이미지 중앙의 수평방향으로의 한 줄만을 이용하므로 계산상의 부하가 크지 않게 된다. 구해진 thresholding값을 기준으로 이보다 큰 값들만을 이용하여 평균(Segment Mean)을 구하고 각 segment를 대표하는 값으로 저장한다.

(2) 구해진 segment 평균값에 대해 auto-thresholding기법을 적용한다.

(3) 2번 과정에서 구한 thresholding 값보다 큰 값을 가지는 segment들의 평균(Segment Mean)들에 대한 전체 평균을 구하여 이미지를 대표하는 값(Image Mode)으로 저장한다.

그림 4~그림 7에 몇 개의 서로 다른 이미지에 대하여 결과를 표와 함께 나타내었다. Image Mode 값이 200이상이면 표식점이 발견되었다고 가정하였다. 그림에서 화살표는 발견된 수직선의 위치를 나타낸다. 마지막 segment에서의 수직선은 실제로 무의미한 값이므로 계산하지 않았다. 그림 4~그림 7의 결과에서 알 수 있듯이 본 실험에서 구현한 이미지 해석 방법은 주어진 표식점을 찾아내는 데 성공적이라고 할 수 있다. 그림 4~그림 6에서는 표식점이 이미지 내에서 차지하는 영역의 크고 작음에 상관없이 Image Mode 값이 200이상의 수치를 나타내고 있다. 이에 비해 그림 7와 같이 표식점이 이미지에 없는 경우에 대해서는 수치가 200이하임을 알 수 있다.

표 1 그림 4에 대한 해석

Table 1 The analysis of Fig. 4

1st Image	Range	Kullback THR.	Line Length	Segment Mean	Image Mode
Segment 0	0-398	253	317	254.2	249.35
Segment 1	399-449	243	307	244.5	
Segment 2	450-491	61	180	61.67	
Segment 3	492-511	62	-	62.67	

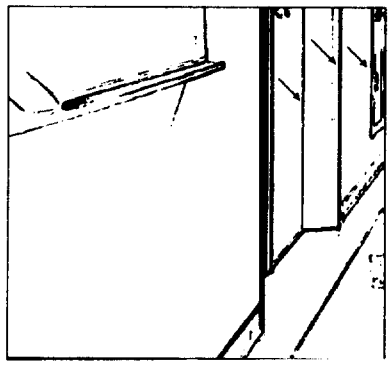

그림 4 제안된 방법의 시험에 사용된 첫번째 복도 이미지

Fig. 4 The first hallway image used to test the proposed method

표 2 그림 5에 대한 해석

Table 2 The analysis of Fig. 5

2nd Image	Range	Kullback THR.	Line Length	Segment Mean	Image Mode
Segment 0	0-111	81	478	154	254.77
Segment 1	112-188	254	478	254.89	
Segment 2	189-352	254	450	254.67	
Segment 3	353-449	70	396	70.5	
Segment 4	450-511	106	-	106.5	

그림 5 제안된 방법의 시험에 사용된 두번째 복도이미지

Fig. 5 The second hallway image used to test the proposed method

표 3 그림 6에 대한해석

Table 3 The analysis of Fig. 6

3rd Image	Range	Kullback THR.	Line Length	Segment Mean	Image Mode
Segment 0	0-147	254	478	254.67	254.67
Segment 1	148-423	103	316	103.5	
Segment 2	424-511	124	-	126.0	

그림 6 제안된 방법의 시험에 사용된 세번째 복도 이미지

Fig. 6 The third hallway image used to test the proposed method

표 4 그림 7에 대한 해석

Table 4 The analysis of Fig. 7

4th Image	Range	Kullback THR.	Line Length	Segment Mean	Image Mode
Segment 0	0-238	116	478	121.33	122.85
Segment 1	239-319	74	478	75.339	
Segment 2	320-511	99	-	124.368	

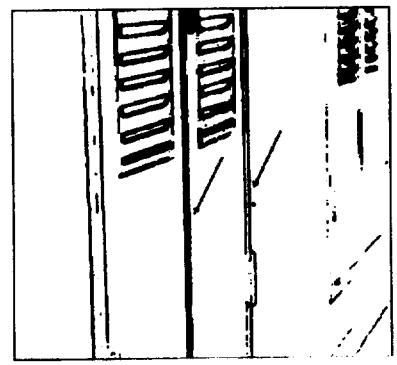

그림 7 제안된 방법의 시험에 사용된 네번째 복도 이미지

Fig. 7 The fourth hallway image used to test the porposed method

한편, 그림 8에 나타낸 로봇 좌표계에서 2차원 평면상의 한 지점($z=0$)에 대한 위치는 식(9)을 이용하여 구할 수 있다.

$$x = 1\frac{z_0 + l_2\sin\phi}{f\sin\phi + \nu\cos\phi}[-f\cos\phi\sin\theta + u\cos\theta + \nu\sin\phi\sin\theta] - l_2\cos\phi\sin\theta$$

$$y = 1\frac{z_0 + l_2\sin\phi}{f\sin\phi + \nu\cos\phi}[f\cos\phi\cos\theta + u\sin\theta - \nu\sin\phi\cos\theta] + l_2\cos\phi\cos\theta$$

$$(z_0 = 1.1m,\ L_2 = 0.04m,\ f = 25mm) \tag{9}$$

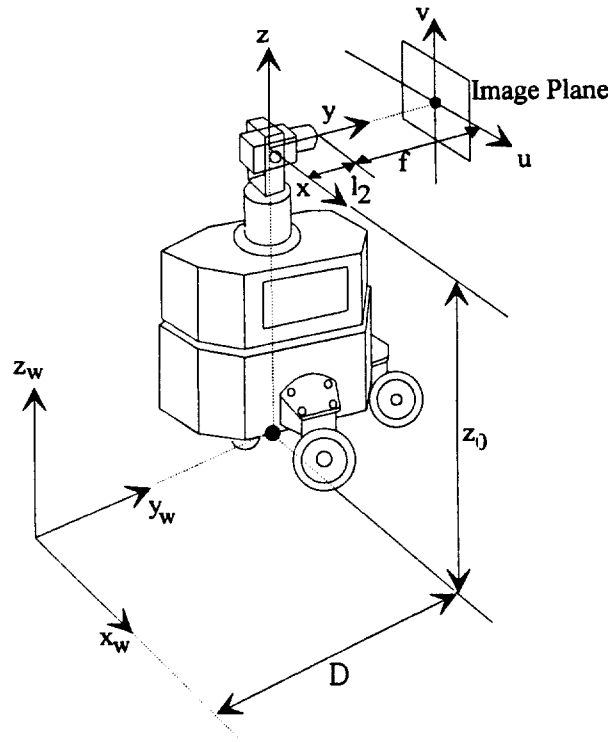

그림 8 로봇 좌표계

Fig. 8 The robot coordinate system

Table 5 Processing time required for one image
표 5 한 이미지를 처리하는데 걸리는 시간

초기화 및 전처리 과정	30sec
이미지 해석	4sec
전체시간	34sec

여기서, z_0는 지면으로부터 스캐너의 중심까지의 높이이고, l_2는 스캐너 중심에서 렌즈 중심까지의 길이이며 f는 촛점거리, ϕ는 tilt angle, θ는 pan angle 을, (u,v)는 이미지상에서의 픽셀 좌표를 나타낸다. 이상 소개한 방법을 이용하여 한 장의 이미지를 처리했을 때 걸리는 시간은 표 5와 같다. 여기서 초기화 과정은 버퍼 할당과 직렬 포트의 초기화를 포함한다. 위의 결과에서 보는 바와 같이 전체 알고리즘에서 주로 low pass filtering 과정과 에지 처리 과정에서 상당한 부분의 시간을 차지하고 있음을 알 수 있다. 이것은 전처리 과정이 주로 이미지 버퍼에 대한 convolution이어서 여기에서 계산량이 많기 때문이다. 이것은 convolution 과정을 보조프로세서를 이용하여 처리할 경우 전체 과정을 15초 이내로 줄일 수 있다.

5.2 시험 주행

본 절에서는 이상의 결과를 바탕으로 로봇으로 하여금 복도를 주행하도록 하였을 때의 결과를 보이기로 한다. 로봇이 복도를 주행함에 있어 주어진 가정으로 로봇은 복도의 수평선과 평행하게 진행한다고 하였다. 로봇의 시험 주행 과정은 다음과 같이 정하였다.

(1) 현 위치에서 표식점이 발견되었을 경우 해당 표식점의 위치와 거리를 계산한다.
(2) 계산된 거리만큼 앞으로 전진한다.
(3) 정지한 후, 새로운 표식점을 찾는다.
(4) 표식점이 발견되었을 경우, 1에서부터 다시 시작한다.
(5) 표식점이 발견되지 않았을 경우, 30cm씩 전진하면서 매 순간마다 표식점을 찾는다.
(6) 4의 과정을 반복한다.

실험은 두 개의 서로 다른 위치에 있는 표식점에 대하여 행하였으며 그림 9와 그림 10, 표 6과 표 7에 실제 주행시 카메라로부터 얻어진 이미지와 결과를 각각 나타내었다. $\bar{x}, \bar{y}$은 실제로 측정한 값을 나타낸 것이고, $\hat{x}, \hat{y}$는 이미지 처리과정을 거쳐 식(9)를 통해 계산된 값이다. 표 7의 θ값은 원래의 θ 값인 16.2° 의 값에서 로봇이 주행하는 동안 로봇의 내부에 있는 엔코더 값을 계속하여 읽어 들여 1.76° 만큼을 원래의 θ값에 보정해 준 것이다. 표 6, 표 7에서 알 수 있듯이 주행 중 로봇과 복도의 평행선과의 어긋난 각을 정확하게 알 수 있다면 y 값에서 큰 오차(약 24cm)가 생기지 않았을 것이다. 즉, 이것은 주행 중 로봇의 예측 불가능한 미끄러짐이나, 정지시의 흔들림 등의 이유로 생기는 문제로서 엔코더 값만으론 보정이 되지 않으며 두 개이상의 표식점을 이용하여 실내 복도에 대한 로봇의 방위각을 구할 수 있다. 그림 11과 그림 12, 표 8과 표 9에 다른 위치에서 새로운 주행을 하였을 때의 이미지와 그 결과를 나타내었다. 각 주행에서 첫 번째 위치는 복도와 평행이 되도록 유지하였으며 카메라에 대한 재조정을 행하였다. 두 번째 주행의 경우에도 두 번째 이미지에서 계산한 위치 값에 비교적

표 6 그림 9에 대한 해석
Table 6 The analysis Fig. 9

1st Image (첫번째 주행)	Range	Line Length	Kullback THR.	Segment Mean	Image Mode
Segment 0	0-65	446	251	254.2	250.85
Segment 1	66-218	368	247	247.5	
Segment 2	219-311	214	61	61.3	
Segment 3	312-511	-	108	109.3	
$\bar{x}$	-1.624m		$\bar{y}$	5.012m	ϕ(deg.)
					-8
$\hat{x}$	-1.5923m		$\hat{y}$	4.4957m	θ(deg.)
					16.2

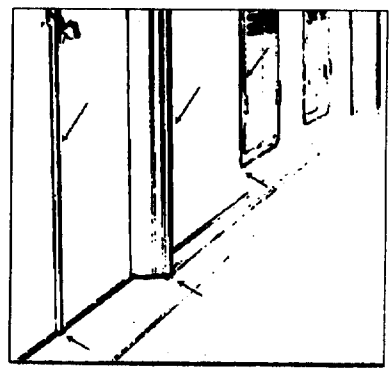

그림 9 첫 번째 시험 주행시의 첫 번째 이미지
Fig. 9 The first image of the first experimental run

표 7 그림 10에 대한 해석
Table 7 The analysis of the Fig. 10

2nd Image (첫번째 주행)	Range	Line Length	Kullback THR.	Segment Mean	Image Mode
Segment 0	0-44	478	89	90.0	249.33
Segment 1	45-142	387	116	120.67	
Segment 2	143-214	375	128	130.0	
Segment 3	215-376	300	243	244.3	
Segment 4	377-467	267	253	254.3	
Segment 5	468-511	-	94	151.5	
$\bar{x}$	-1.31m		$\bar{y}$	6.633m	ϕ(deg.)
					-8
$\hat{x}$	-1.2756m		$\hat{y}$	6.8688m	θ(deg.)
					17.96

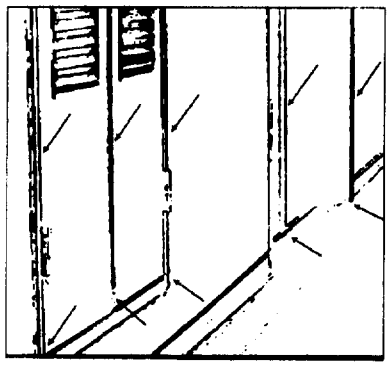

그림 10 첫 번째 시험 주행시의 두 번째 이미지
Fig. 10 The second image of the first experimental run

표 8 그림 11에 대한 해석
Table 8 The analysis of Fig. 11

1st Image (두번째 주행)	Range	Line Length	Kullback THR.	Segment Mean	Image Mode
Segment 0	0-234	400	254	254.5	253.5
Segment 1	235-352	345	252	252.5	
Segment 2	353-511	-	80	144.5	
$\bar{x}$	-1.19m		$\bar{y}$	5.41m	ϕ(deg.)
					-8
$\hat{x}$	-1.1398m		$\hat{y}$	5.3988m	θ(deg.)
					15.2

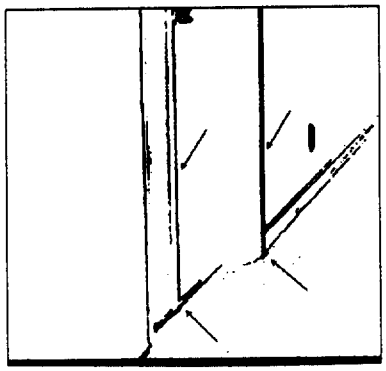

그림 11 두 번째 시험 주행시의 첫 번째 이미지
Fig. 11 The first image of the second experimental run

표 9 그림 12에 대한 해석
Table 9 The analysis of Fig. 12

2nd Image (두번째 주행)	Range	Line Length	Kullback THR.	Segment Mean	Image Mode
Segment 0	0-283	386	61	65.0	254.67
Segment 1	284-365	278	254	254.67	
Segment 2	366-408	276	81	81.5	
Segment 3	409-511	-	101	114.21	
$\bar{x}$	-1.49m		$\bar{y}$	6.5675m	ϕ(deg.)
					-8
$\hat{x}$	-1.6117m		$\hat{y}$	6.503m	θ(deg.)
					17.66

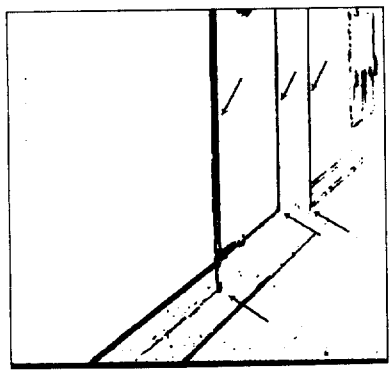

그림 12 두 번째 시험 주행시의 두 번째 이미지
Fig. 12 The second image of the second experimental run

큰 오차(x 방향으로 12.17cm)가 생기고 있음을 알 수 있다. 이것은 로봇이 주행하다가 정지하는 순간 얼마간 회전하는 경우가 있고, 실제로 회전하는 경우에도 주어진 명령만큼 정확하게 추종하지 않기 때문이다. 또한, 바닥에서 주행 중 미끄러지는 경우도 생길 수 있으므로 캘리브레이션된 카메라로 위치를 정확하게 측정하더라도 예측할 수 없는 형태의 오차를 유발하게 된다. 즉, 시험 주행에서의 가정을 위반하고 있다. 이 외에 식 (9)에서 알 수 있듯이 카메라의 외부 파라메타(θ, ϕ)가 부정확할 경우 x, y의 값이 큰 영향을 받게 되므로 카메라 동작이 포함되는 경우, 로봇의 주행중에도 계속적인 외부 파라메타 측정이 필요함을 알 수 있다. 한편, 표 9의 segment 2의 평균이 표식점임에도 불구하고 81이라는 상당히 낮은 수치를 나타내고 있다. 이것은 시험 주행시의 조명 상태의 변화에 따른 색 정보의 변화에 기인하는 것으로 보여지며 본 실험에서는 이에 대한 보정은 행하지 않았다. 이에 대한 해결책은 hue, saturatoin 이미지에 대한 에지 처리등의 방법을 통하여 조명의 효과를 상쇄시킬 수 있는 방법을 강구하여야 할 것이다.

6. 결 론

본 연구에서 표식점의 컬러 정보를 이용하여 물체 인식을 쉽게 할 수 있음을 보였다. 그리고 컬러 정보를 다룰 수 있는 thresholding 방법을 소개하였으며 컬러 정보에 의한 데이타량의 증가에도 불구하고 실제 알고리즘에서 차지하는 계산 시간을 늘이지 않으면서 컬러 정보를 이용할 수 있음을 보였다. 만약, 흑백 이미지만을 이용하여 같은 내용의 실험을 하고자 하는 경우, 찾고자 하는 표식점의 위치를 이미지의 공간적인 구조를 통하여 구하지 않으면 안된다. 예를 들어, Hough transfrom을 이용하여 표식점을 찾고자 하는 경우, 발견된 여러 수직선들 중 어느 것이 표식점에 해당하는 것이냐에 대한 correspondence 문제에 봉착하게 된다. 반면, 컬러 정보를 이용하였을 경우, 간단한 방법으로 표식점을 감지할 수 있었으며 또한 비교적 다루기 쉬운 수직선을 이용하여 바닥과의 교차점을 구할 수 있었다. 찾고자 하는 수직선의 위치도 표식점의 saturation 정보를 이용하여 여러 개의 수직선 중 어느 것인가를 파악할 수 있었으며 컬러 정보를 이용하는 방법도 hue나 saturation 버퍼 전부를 이용하는 것이 아니라 saturation 버퍼 내의 512 바이트에 대한 계산만을 행하기 때문에 전체 알고리즘을 구현함에 있어 큰 문제가 되지 않으며 이것은 correspondence 문제를 해결함에 있어 컬러 정보를 사용하는 것이 훨씬 수월함을 의미한다. 하지만, 컬러 정보는 현재 로봇이 위치한 실내에 대한 어느 정도의 지식을 가지고 있어야 가능하며 실제로 로봇이 임의의 환경에 적응하기 위해서는 intensity 이미지를 먼저 이용하여 공간적인 구조를 파악한 후 컬러 정보를 파악하는 과정을 도입하여야 할 것이다.

참 고 문 헌

[1] K. S. Fu, R. C. Gonzalez and C. S. G.Lee, ROBOTICS: Control, Sensing, Vision, and Intelligence, McGraw-Hill, 1987.

[2] Nicholas Ayache, Artificial Vision for Mobile Robots : Stereo Vision and Multisensory Perception, The MIT Press, 1991.

[3] R. O. Duda and P. E. Hart, Pattern Classification and

Scene Analysis, New York:Wiley-Interscience, 1973.
[4] Kokichi Sugihara, "Some Location Problems for Robot Navigation Using a Single Camera," CVGIP 42, pp.112-129, 1988.
[5] Takeshi Shakunaga, "3-D Corridor Modeling from a Single View under Natural Lighting Conditions," IEEE Trans. PAMI. vol. 14, no. 2, pp.293-298, Feb. 1992.
[6] Glenn Healey, "Color Discrimination By Computer," IEEE Trans. SMC. vol. 19, no. 6, pp.1613-1617, Nov. 1989.
[7] Douglas C. Brockelbank, "An Experimental Investigation in the Use of Color in Computational Stereopsis," IEEE Trans. SMC. vol. 19, no. 6, pp.1365-1383, Nov. 1989.
[8] F. Bumbaca & K.C.Smith, "Design and Implementation of a Colour Vision Model for Computer Vision Applicaiotions," CVGIP 39, pp.226-245, 1987.
[9] Nobuyuki Otsu, "A Threshold Selection Method from Gray-Level Histograms," IEEE Trans. SMC. vol. 9, no. 1, pp.62-66, Jan. 1979.
[10] R. M. Haralick, L. G. Shapiro, Computer and Robot Vision, Addison-Wesley, 1992.
[11] William K.Pratt, Digital Image Processing, Wiley Interscience, 1991.
[12] 방 석원, 펄스위상차와 Structured light를이용한 이동 로봇 시각 장치 구현, KAIST 석사 학위 논문, 1991.
[13] K2A Mobile Platform User's Manual, Cybermation Inc. 1989

저 자 소 개

유 원 필 (兪 元 弼)

1968년 12월 27일생. 1992년 2월 서울대 공대 제어계측공학과 졸업. 1994년 2월 한국과학기술원 전기 및 전자공학과 졸업(석사). 현재 한국과학기술원 전기 및 전자공학과 박사과정

정 명 진 (鄭 明 振)

1950년 1월 31일생. 1973년 서울대 공대 전기공학과 졸업. 1977년 미국 미시간 대학교 ECE 졸업. 1983년 동 대학원 CICE졸업(공박). 1976년 국방과학연구소 연구원, 1981년~1983년 미시간 대학교 CRIM연구조교. 1983년~현재 한국과학기술원 전기 및 전자공학과 교수